Claims

[c1] 1. A method for operating a processor having an architecture of a larger bitlength with a program comprising instructions compiled to produce instruction results of at least one smaller bitlength, characterized by the steps of:

detecting when in program order a first smaller bitlength instruction is to be dispatched which does not have the target register address as one of its sources;

adding a so-called "extract" instruction into an instruction stream before the smaller bitlength instruction, the extract instruction comprising the following steps of:

a. dispatching the extract instruction together with the following smaller bitlength instruction from an instruction queue into a Reservation Station;

b. issuing the extract instruction to an Instructional Execution Unit (IEU) as soon as all source operand data is available and an IEU is available according to respective issue scheme;

c. executing the extract instruction by an available IEU;

d. setting an indication that the result of said instruction needs to be written into the result field of the instruction following the extract instruction, and;

e. writing the extract instruction result into the result field of said first instruction, and into all fields of operands being dependent of said first instruction.

[c2] 2. The method according to claim 1 in which the step of writing the extract instruction result into the result field of said first instruction, is controlled by incrementing a tag specifying in which location the result of the first instruction has to be written.

[c3] 3. The method according to claim 1, further comprising the steps of having the larger bitlength equal to 64-bit and the smaller bitlength equal to 32-bit.

[c4] 4. The method according to claim 1, further comprising the steps of having the larger bitlength equal to 128-bit and the smaller bitlength equal to 64 or 32-bit.

[c5] 5. The method according to claim 1, further comprising the step of when said second instruction is dependent of the first instruction, selectively inserting an extract instruction.

[c6] 6. The method according to claim 1, further comprising the steps of dispatching said first instruction in the same cycle as the extract instruction, and assuring that in the same cycle both or none of said two instructions is written into a reservation station means, and in case of a multiple write into the same result latch reducing the respective multiple input signals for the latch by either one of an OR gate or an AND gate, respectively.

[c7] 7. The method according to claim 1, further comprising the step of associating the same instruction execution unit for said first and said extract instruction.

[c8] 8. A computer system having an out-of-order processing system for operating a processor having an architecture of a larger bitlength with a program comprising instructions compiled to produce instruction results of at least one smaller bitlength, said computer system uses a computer readable machine language, said computer readable machine language comprises:
a first computer readable code for detecting when in program order a first smaller bitlength instruction is to be dispatched which does not have a target register address as one of its sources;
a second computer readable code for adding a so_extract_ instruction into an instruction stream before the smaller bitlength instruction, said extract instruction comprising additional computer readable code, said additional computer readable code comprises:
a third computer readable code for dispatching the extract instruction together with the following smaller bitlength instruction from the instruction queue into a Reservation Station,
a fourth computer readable code for issuing the extract instruction to an Instruction Execution Unit (IEU) as soon as all source operand data is available and an IEU is available according to respective issue scheme,
a fifth computer readable code for executing the extract instruction by an available IEU,

a sixth computer readable code for setting an indication that the result of said instruction needs to be written into the result field of the instruction following the extract instruction, and

a seventh computer readable code for writing the extract instruction result into the result field of said first instruction, and into all fields of operands being dependent of said first instruction.

[c9] 9. A computer system, said computer system executes a readable machine language, said readable machine language comprises:

a first computer readable code for detecting when in program order a first smaller bitlength instruction is to be dispatched which does not have a target register address as one of its sources;

a second computer readable code for adding a so_extract_ instruction into an instruction stream before the smaller bitlength instruction, said extract instruction comprising additional computer readable code, said additional computer readable code comprises:

a third computer readable code for dispatching the extract instruction together with the following smaller bitlength instruction from the instruction queue into a Reservation Station,

a fourth computer readable code for issuing the extract instruction to an Instruction Execution Unit (IEU) as soon as all source operand data is available and an IEU is available according to respective issue scheme,

a fifth computer readable code for executing the extract instruction by an available IEU,

a sixth computer readable code for setting an indication that the result of said instruction needs to be written into the result field of the instruction following the extract instruction, and

a seventh computer readable code for writing the extract instruction result into the result field of said first instruction, and into all fields of operands being dependent of said first instruction.

[c10] 10. A processing system having means for executing a readable machine language, said readable machine language comprises:

a first computer readable code for detecting when in program order a first

smaller bitlength instruction is to be dispatched which does not have a target register address as one of its sources;

a second computer readable code for adding a so_extract_ instruction into an instruction stream before the smaller bitlength instruction, said extract instruction comprising additional computer readable code, said additional computer readable code comprises:

a third computer readable code for dispatching the extract instruction together with the following smaller bitlength instruction from the instruction queue into a Reservation Station,

a fourth computer readable code for issuing the extract instruction to an Instruction Execution Unit (IEU) as soon as all source operand data is available and an IEU is available according to respective issue scheme,

a fifth computer readable code for executing the extract instruction by an available IEU,

a sixth computer readable code for setting an indication that the result of said instruction needs to be written into the result field of the instruction following the extract instruction, and

a seventh computer readable code for writing the extract instruction result into the result field of said first instruction, and into all fields of operands being dependent of said first instruction.